REMARKS/ARGUMENTS

Claims 1 - 16 are pending.

Distinctions between the Morrison et al. reference and the pending claims was required. Claims 1 and 5 have been amended to correct minor informalities, and thus have not substantively changed the scope of the claims as previously presented.

Independent claim 1

The present invention is directed to processors. As recited in independent claim 1, a scalable processor having general registers and a general purpose memory includes "a memory device" operative to store program instructions in definable locations. The program instructions propagate through the memory device, and the positions of the instructions in the memory is associated with a timetag. A "plurality of processing elements" are distributed throughout the memory device and are able to receive program instructions from the current positions in the memory device, wherein each of the processing elements executes an instruction in its current position without regard to order. A "plurality of active stations" are associated with the processing elements for enforcing programmatic order as indicated by the timetag.

Morrison et al. show in Fig. 6 a system architecture (600) having a memory subsystem (610) interconnected to a plurality of logical resource drivers (LRDs, 620) over a network (630). The logical resource drivers (620) are further interconnected to a plurality of processor elements (640) over a network (650). The plurality of processor elements (640) are interconnected over a network (670) to the shared resources containing a pool of register set and condition code set files (660). Any processor element (640) can access any register file or condition code storage in any context file (660). Any processor element (640) can access any logical resource driver (620) and any logical resource driver (620) can access any portion of the memory subsystem (610). *Col. 15, lines 5 - 26.*

As mentioned above, the present invention is directed to a processor having a general purpose memory. An aspect of the present invention as recited in claim 1 is "a memory device" to store instructions, and through which processing elements are distributed. An illustrative embodiment of this aspect of the invention is shown in Figs. 2 and 7 - 9 of the instant

Appl. No. 09/828,600
Amdt. sent November 10, 2004
Reply to Office Action of October 21, 2004

PATENT

specification. Fig. 2 shows an architecture having a main memory and a "memory device" identified in the figure as an instruction window (see also Fig. 7). The instruction window comprises sharing groups, each having active stations (AS). As can be seen in Fig. 9, each active station includes an instruction register (INSTR) for storing an instruction. Kindly refer to page 20, line 11 of the specification. Fig. 8 shows that each sharing group is associated with a processing element (PE). The figures therefore illustrate an embodiment of a "memory device" and "processing elements" distributed through the memory device. Fig. 7 illustrates an embodiment of the recited "memory device" comprising an arrangement of sharing groups where the processing elements are distributed through the memory device.

Referring to Morrison et al. for comparison, they show only a memory (610); they do not show an element that can be fairly construed as Applicants' recited "memory device" having "processing elements" distributed through the memory device. Firstly, Morrison et al. show only a memory subsystem (610). Secondly, though the Morrison et al. LRD's (620) include storage for instructions, they do not show a distribution of processing elements through the LRD's in the manner recited in the pending claims. For example, it can be clearly seen in Fig. 6 that their processor elements (640) are not distributed through the LRD's (620). In fact, their processing elements (640) are cross-coupled via a network (650) to the LRD's (620). Morrison et al. do not show this aspect of the present invention.

Another aspect of the present invention as recited in claim 1 is that the position of an instruction in the memory device is associated with a timetag. This aspect of the present invention is described in the specification beginning on page 13, line 12. It is earnestly submitted that Morrison et al., as understood, do not disclose anything in their architecture that could fairly be construed as Applicants' recited timetag. Morrison et al. describe the use of a logical processor number and an instruction firing time (IFT) to control the concurrent execution of multiple instructions. *Col. 4, lines 7 - 15.* The instruction firing time is programmatically determined (via their TOLL software) by analyzing the instruction stream and considering the temporal usage of each resource required by the instruction. *Id and col. 10, line 39.* By comparison, Applicants' claimed timetag is associated with position of an instruction in the memory device; e.g., "position of each of said executable program instructions in said memory

device is associated with a current timetag." *Claim 1*. Morrison et al. teach an IFT that is programmatically determined and thus do not show a timetag that is associated with position of an instruction in a memory device. Moreover, Morrison et al., as discussed above, do not show a memory device in to begin with, and for at least this reason alone do not show Applicants' claimed timetag.

Another aspect of the present invention as recited in claim 1 is "a plurality of active stations" associated with each of the processing elements for enforcing programmatic order as indicated by the timetag. This aspect of the invention is shown by the example in Fig. 9 and discussed beginning on page 26, line 20. First, as discussed above, Morrison et al. do not describe "timetags" as recited in the pending claims and so the reference cannot be construed to show an active station that enforces programmatic order as indicated by the timetag. Moreover, Morrison et al. disclose the use of a program they refer to as TOLL to programmatically determine order of execution. It is submitted with earnest that their TOLL program does not constitute a plurality of active stations associated with each of the processing elements. Morrison et al. do not show this aspect of the present invention as recited in the pending claims.

Still another aspect of the present invention, as recited in claim 1, is that each of the processing elements executes an instruction in its current position in the memory device without regard to order. By comparison, Morrison et al. uses software (namely, TOLL) to programmatically insert instruction firing time (IFT) information in the instruction stream. *Col. 4, lines 4 - 14*. Instructions in the Morrison et al. processor are execute according to the IFT. Morrison et al. therefore cannot be construed as showing the execution of instructions without regard to order.

Independent claim 5

As recited in independent claim 5, a scalable processor having general registers and a general purpose memory includes "a memory device" to store program instructions in definable locations, where each program instruction propagates through the memory device. Each position of a program instruction in the memory device is associated with a current timetag. A "plurality of processing elements" are distributed throughout the memory device, each

receiving an executable program instruction from a current position of the memory device and executes the instruction without regard to order. A "plurality of active stations" are associated with each of the processing elements to enforce programmatic ordering of the instructions as indicated by the state of said timetag. Each of the active stations includes a dedicated timetag register for capturing a temporally closest previous broadcast timetag value for comparison with a timetag of a datum sharing a common address in the active station, in order to accomplish at least one of the following: a) enforcement of programmatic ordering, b) linking an instruction with a closest previous related instruction as indicated by a common memory address in the general purpose random access memory, a predicate address, or a register address in the general purpose register, or c) minimizing dependencies among instructions.

In addition to the distinctions discussed above, between the present invention and the cited art, claim 5 further distinguishes over Morrison et al. with respect to each of the active stations having a timetag register for capturing a temporally closest previous broadcast timetag value. As discussed above, Morrison et al. use their TOLL software to programmatically analyze the instruction stream to enhance the instruction stream with added intelligence such as logical processor numbers and instruction firing times. *Col. 4, lines 4 et seq.* By so doing, they can assign instructions to their processor elements (640) to achieve concurrent execution of multiple instructions. Morrison et al. therefore cannot be construed to show a plurality of active stations, each having a timetag register, as recited in claim 5. Morrison et al. teach a program which adds intelligence to the program stream, and by so doing achieve proper order of execution among concurrently executing instructions. Applicants' claim 5 recites a plurality of active stations to accomplish at least one of the following: a) enforcement of programmatic ordering, b) linking an instruction with a closest previous related instruction as indicated by a common memory address in the general purpose random access memory, a predicate address, or a register address in the general purpose register, or c) minimizing dependencies among instructions. Morrison et al. do not show this aspect of the present invention.

Independent claim 6

Independent claim 6 recites a general purpose processing system having a plurality of sharing groups, a plurality of processing elements each being associated with at least one sharing group, a memory device for storing program instructions to be executed, and a plurality of active stations which form one of the sharing groups. The general purpose processing system further includes a plurality of spanning bus segments arranged in columns, and a forwarding unit associated with each sharing group. Each forwarding unit in a column is coupled to monitor its corresponding level in an adjacent spanning bus segment, and is coupled to each sharing group in the column. Each of the processing elements receives an instruction from the memory device and executes the instruction as soon as its operand is acquired, whereby a plurality of executable program instructions are executed in parallel during each instruction cycle.

In addition to the distinctions between the present invention and the cited art, discussed above, claim 6 further distinguishes over Morrison et al. with respect to the recited spanning buses. The spanning buses are arranged in columns and there are sharing groups in a column. This aspect of the present invention is shown in the example of Fig. 7, as described in the specification beginning on page 14, line 12. By comparison, Morrison et al. use networks (630, 650, 670) to cross-connect their components. None of these networks can be construed as spanning buses arranged in columns; their networks "are full access networks that could be composed of conventional crossbar networks, omega networks, banyan networks, or the like." *Col. 15, lines 15 - 18.* Morrison et al. therefore do not show Applicants' spanning buses.

Independent claim 10

Independent claim 10 recites a scalable processing system having a memory device that is operative to store a plurality of executable program instructions in definable locations, each instruction being propagated through the memory device. The scalable processing system further includes a plurality of processing elements, each being associated with an output result that is produced as a result of executing one of the instructions. The scalable processing system further includes a plurality of active stations, each holding a single program

instruction, each being associated with at least one of the processing elements. The scalable processing system further includes a spanning bus structure configured to couple at least some output results associated with the processing elements to at least some of the active stations, each output result having a timetag and an address associated with it. Each active station has at least one input having a timetag and an address associated with it; the input being associated with execution of an instruction. Each active station includes first comparison logic, second comparison logic, and firing logic. The first comparison logic compares the timetag associated with its input with the timetag associated with an output result on the spanning bus, to produce a first comparison result. The second comparison logic compares the address associated with its input with an address associated with an output result on the spanning bus, to produce a second comparison result. The firing logic can issue a given instruction to one of the processing elements multiple times, each time being based on the first comparison result and the second comparison result.

In addition to the distinctions between the present invention and the cited art, discussed above, claim 10 further distinguishes over Morrison et al. with respect to the recited detail in the active stations. Claim 10 recites first and second comparison logic and firing logic which issues instructions based on results of the first and second comparison logic. Morrison et al. control instruction firing based on logical processor numbers and instruction firing times, which are determined by programmatic analysis (the TOLL software) of the instruction stream. It is earnestly submitted that their TOLL software does not show the recited details of Applicants' active stations. Consequently, Morrison et al. do not show this aspect of the present invention.

Appl. No. 09/828,600
Amdt. sent November 10, 2004
Reply to Office Action of October 21, 2004

PATENT

CONCLUSION

In view of the foregoing, all claims now pending in this Application are believed to be in condition for allowance. The issuance of a formal Notice of Allowance at an early date is respectfully requested.

If the Examiner believes a telephone conference would expedite prosecution of this application, please telephone the undersigned at 650-326-2400.

Respectfully submitted,

George B. F. Yee
Reg. No. 37,478

TOWNSEND and TOWNSEND and CREW LLP
Two Embarcadero Center, Eighth Floor
San Francisco, California 94111-3834
Tel: 650-326-2400
Fax: 415-576-0300
GBFY:cmm
60344014 v1